Subject: Re: what is the best way to do a surface (or 2D) interpolation?
Posted by Vince Hradil on Tue, 23 Sep 2008 19:46:30 GMT
View Forum Message <> Reply to Message

On Sep 23, 12:25 pm, Brian Larsen <balar...@gmail.com> wrote:
>> Here's a way to get verts:
>
>> sz = size(array)
>> nx = sz[0]
>> ny = sz[1]
>> nz = sz[2]
>> ns = sz[sz[0]+2]
>> verts = findgen(ns)
>> verts = transpose([[verts mod nx], [verts/nx mod ny], [verts/nx/
>> ny]])
>
>> BTW, I'd like to find a faster way, if there is one.
>
> This looks like the right thing but doesn't seem to give the right
> answer (or am I using it wrong?)
>
> ;; this is my data
> array=findgen(15,3)
> ;; and get the verts
> sz = size(array)
> nx = sz[0]
> ny = sz[1]
> nz = sz[2]
> ns = sz[sz[0]+2]
> verts = findgen(ns)
> verts = transpose([[verts mod nx], [verts/nx mod ny], [verts/nx/
> ny]])
>
> IDL> print, verts
> 0.00000 0.00000 0.00000
> 1.00000 0.500000 0.0333333
> 0.00000 1.00000 0.0666667
> 1.00000 1.50000 0.100000
> 0.00000 2.00000 0.133333
> 1.00000 2.50000 0.166667
>
> The z values that are here are not in my original array...
>
> Sorry to hijack your post Paula,
>
> Brian
>
> ---------- ----------

> Brian Larsen
> Boston University
> Center for Space Physicshttp://people.bu.edu/balarsen/Home/IDL

My example is for 3d arrays only! For 2d arrays, just use:
nx = sz[1]
ny = sz[2]
verts = lindgen(nx*ny)
verts = transpose([[verts mod nx], [verts/nx]])

There was a typo in the original, too. nx=sz[1] not sz[0], and so on.

Yes, it should be lindgen...
